Approximate computations with modular curves

Jean-Marc Couveignes* and Bas Edixhoven

Abstract. This article gives an introduction for mathematicians interested in numerical computations in algebraic geometry and number theory to some recent progress in algorithmic number theory, emphasising the key role of approximate computations with modular curves and their Jacobians. These approximations are done in polynomial time in the dimension and the required number of significant digits. We explain the main ideas of how the approximations are done, illustrating them with examples, and we sketch some applications in number theory.

2010 Mathematics Subject Classification. Primary 65-D-99; Secondary 11-Y-40, 14-Q, 11-F-80, 11-G-18, 14-G-35, 14-G-40.

Keywords. Drinfeld modules, L-functions, Weil conjecture.

1. Introduction

The purpose of this article is to give an introduction to the main results of the book [BCEJM] and their generalization in the PhD thesis [Bru1] and in [Bru2], as well as some applications, and most of all to explain the essential role played by *approximate computations*. The intended reader is a mathematician interested in *numerical computations* in algebraic geometry or number theory.

The results concern fast algorithms in number theory and more precisely, fast computation of Fourier coefficients of modular forms. These coefficients, with Ramanujan's τ-function as a typical example, have deep arithmetic significance and are important in various areas of mathematics, from number theory and algebraic geometry to combinatorics and lattices.

The fastest previously known algorithms for computing these Fourier coefficients took exponential time, except in some special cases. The case of elliptic curves (Schoof's algorithm) was at the birth of elliptic curve cryptography around 1985. The results mentioned above give an algorithm for computing coefficients of modular forms in polynomial time. For example, Ramanujan's $\tau(p)$ with p a prime number can be computed in time bounded by a fixed power of $\log p$.

Such fast computation of Fourier coefficients is itself based on the main result of the book: the computation, in polynomial time, of Galois representations over finite fields attached to modular forms by the Langlands program.

The computation of the Galois representations uses their realisation, following Shimura and Deligne, in the torsion subgroup of Jacobian varieties of modular curves. The main challenge is then to perform the necessary computations in time polynomial in the dimension of these nonlinear algebraic varieties. Exact computations involving systems of polynomial equations in many variables take exponential time. This is avoided by numerical approximations with a precision that suffices to derive exact results from them. Bounds for the required precision – in other words, bounds for the height of the rational numbers that describe the Galois representation to be computed – are obtained from Arakelov theory.

This article is organised as follows. Sections 2 and 3 are concerned with numerical methods used in the context of complex algebraic curves and their Jacobian varieties. Sections 4 and 5 describe how to get exact results about torsion points on modular curves using these numerical methods. Section 4 focuses on the genus 1 curve X_{11} while Section 5 deals with the general modular curve X_ℓ. As an application, Section 6 gives two examples of fast computation of coefficients of modular forms: Ramanujan's τ-function, and the classical sums of squares problem.

*Research supported by ANR (project ALGOL ANR-07-BLAN-0248) and by DGA maîtrise de l'information.

2. Algorithms for curves and Jacobians

Let X be a connected, smooth, projective algebraic curve over the field $\mathbb{C}$ of complex numbers. The set $X(\mathbb{C})$ of complex points of X is a Riemann surface. Let g be the genus of X and let $(\omega_k)_{1\leq k\leq g}$ be a basis for the space of holomorphic differentials on X. We fix a point $b \in X(\mathbb{C})$ and we denote by Y_b the set of homotopy classes of paths on $X(\mathbb{C})$ starting at b. The *universal cover* $f_b : Y_b \to X(\mathbb{C})$ maps every path to its end point. The *fundamental group* $\pi_1(X(\mathbb{C}), b) \subset Y_b$ is the subset of (homotopy classes of) closed paths. It acts on Y_b, with quotient $X(\mathbb{C})$. We have an integration map $\phi_b : Y_b \to \mathbb{C}^g$ defined by

$$\phi_b(\gamma) = (\int_\gamma \omega_1, \ldots, \int_\gamma \omega_g).$$

The image of $\pi_1(X(\mathbb{C}), b)$ by ϕ_b is a lattice Λ in $\mathbb{C}^g$. It is called the *lattice of periods*. It is a free $\mathbb{Z}$-module of rank $2g$. The quotient $\mathbb{C}^g/\Lambda$ is a complex torus. It is the set of complex points $J(\mathbb{C})$ on the *Jacobian variety* J of X. The integration map $\phi_b : Y_b \to \mathbb{C}^g$ induces a map between the quotients $X(\mathbb{C}) \to J(\mathbb{C})$. This map is a morphism of varieties $X \to J$. We call this morphism ϕ_b also. For every positive integer k we denote $\phi_b^k : X^k \to J$ the morphism that maps $(P_1, \ldots, P_k)$ onto $\phi_b(P_1) + \cdots + \phi_b(P_k)$. Since the image in J does not depend on the ordering on the points P_j, we write $X^{(k)}$ for the k-th symmetric power of X. We note that $X^{(k)}$ is the quotient of X^k by the action of the symmetric group. It is a nonsingular variety. We define the morphism $\phi_b^{(k)} : X^{(k)} \to J$ that maps $\{P_1, \ldots, P_k\}$ onto $\phi_b(P_1) + \cdots + \phi_b(P_k)$. For $k = g$ the map $\phi_b^{(g)}$ is birational and surjective. It is not an isomorphism unless $g \leq 1$. Its fibers are projective linear spaces, mostly (but not all) points. A degree g effective divisor $P = P_1 + \cdots + P_g$ is said to be *non-special* if the map $\phi_b^{(g)}$ is a local diffeomorphism at P. Otherwise we say that P is *special*. This definition does not depend on the chosen origin b. The set of special effective degree g divisors is the singular locus of $\phi_b^{(g)}$. All these maps $\phi_b^{(k)}$ are called Abel-Jacobi maps. In particular

$$\phi_b^{(g)}(\{P_1, \ldots, P_g\}) = \sum_{1\leq j\leq g} (\int_b^{P_j} \omega_k)_k \mod \Lambda,$$

where we can integrate $\int_b^{P_j} \omega_k$ along any path between b and P_j, provided we keep the same path for all k. We can apply the Abel-Jacobi map to any divisor on X. We set $\phi_b(\sum_j e_j P_j) = \sum_j e_j \phi_b^{(1)}(P_j)$. We note that for degree zero divisors, the image does not depend on the origin b. A divisor is said to be *principal* if it is the divisor of a non-zero meromorphic function on X. Two divisors are said to be *linearly equivalent* when their difference is principal. Any principal divisor has degree zero. A degree zero divisor is principal if and only its image by ϕ_b is zero. So the set $J(\mathbb{C}) = \mathbb{C}^g/\Lambda$ of complex points on the Jacobian is canonically identified with the group $\mathrm{Pic}^0(X)$ of linear equivalence classes of degree zero divisors on X.

We now list important algorithmic problems related to the Abel-Jacobi map. We illustrate them on the simple example of the projective curve X with equation

$$Y^2Z - YZ^2 = X^3 - X^2Z. \tag{2.0.1}$$

This curve has genus 1. We write $x = X/Z$ and $y = Y/Z$. The unique (up to a multiplicative constant) holomorphic differential on X is

$$\omega = \frac{dx}{2y-1} = \frac{dy}{x(3x-2)}.$$

We choose the point $b = [0 : 1 : 0]$ as origin for the integration map. For every computational problem we shall consider, we will also explain what can be proven when X is a modular curve X_ℓ and ℓ (therefore g) tends to infinity. The definition of the modular curve X_ℓ is given in Section 5. See also textbooks [Di-Sh, Ste] where X_ℓ is often denoted $X_1(\ell)$.

2.1. Computing the lattice of periods. We first need a basis for the singular homology group $H_1(X(\mathbb{C}), \mathbb{Z})$. If X is the genus one curve given by equation (2.0.1), such a basis can be deduced from the study of the degree two map $x : X \to \mathbb{P}^1$ that sends (x, y) onto x and $[0 : 1 : 0]$ to ∞. This map is ramified at ∞ and the three roots of $4x^3 - 4x^2 + 1$. We lift a simple loop around ∞ and one of these three roots. We then lift a simple loop around ∞ and another root. We thus obtain two elements in $H_1(X(\mathbb{C}), \mathbb{Z})$ that form a basis for it.

Integrating a differential along a path is easy. We express the differential in terms of local coordinates. We then reduce to integrating converging power series. We integrate term by term. In case X is the curve given in equation (2.0.1), we obtain a basis (Ω_1, Ω_2) for the lattice Λ of periods where

$$\begin{aligned}\Omega_1 &= 6.346046521397767108443973084,\\ \Omega_2 &= -3.173023260698883554221986542 + 1.458816616938495229330889613i.\end{aligned}$$

These calculation are made e.g. using the [PARI] system.

```
>a1=0;a2=-1;a3=-1;a4=0;a6=0;
>X=[a1,a2,a3,a4,a6];X=ellinit(X);
>X.omega
[6.346046521397767108443973084,
-3.173023260698883554221986542 + 1.458816616938495229330889613*I]
```

When dealing with general modular curves, an explicit basis for both the singular homology and the de Rham cohomology is provided by the theory of Manin symbols [Man, Merel, Cre, Fre, Ste]. Computing (good approximations of) periods is then achieved in time polynomial in the genus and the required accuracy [Cou2]. The practical side is described in [Bos1, §6.3]. Textbooks [Coh], [Cre, Chapter 3] give even faster techniques for genus 1 curves, but we shall not need them.

2.2. Computing with divisor classes. A degree zero divisor class can be represented by a point in the torus $\mathbb{C}^g/\Lambda = J(\mathbb{C})$. It can also be represented by a divisor of the form

$$P_1 + \cdots + P_g - gb \tag{2.2.1}$$

in this class. This latter representation is not always unique. It is however unique for most classes because $\phi_b^{(g)}$ is birational. The addition problem in this context is the following: given two degree g effective divisors $P = P_1 + \cdots + P_g$ and $Q = Q_1 + \cdots + Q_g$, one would like to compute a degree g effective divisor $R = R_1 + \cdots + R_g$ such that the divisor class of $R - gb$ is the sum of the divisor classes of $P - gb$ and $Q - gb$. So we look for g complex points $R_1, \ldots, R_g$ such that $P_1 + \cdots + P_g + Q_1 + \cdots + Q_g - 2gb$ is linearly equivalent to $R_1 + \cdots + R_g - gb$. This is achieved using the Brill-Noether algorithm [Bri-Noe, Vol]. This algorithm uses a complete linear space $\mathcal{L}$ of forms or functions. This space should have dimension $\geq 2g + 1$. For example, assuming $g \geq 4$, we may take for $\mathcal{L}$ the space of all holomorphic quadratic differential forms. We compute once for all a basis for this space. Then the Brill-Noether algorithm alternates several steps of two different natures. Sometimes we are given a form (function) and we want to compute its divisor. Sometimes we are given an effective divisor D and we want to compute a basis for the subspace $\mathcal{L}(-D)$ consisting of forms (functions) vanishing at this divisor.

The first problem (finding zeros of a given form) can be reduced, using a convenient coordinate system, to the following problem: given a power series $f(z) = \sum_{k\geq 0} f_k z^k$ with radius of convergence ≥ 1, find approximations of its zeros in the disk $D(0, 1/2)$ with center 0 and radius 1/2. It is clear (see [Cou1, §5.4]) that, for the purpose of finding zeros, one can replace $f(z)$ by its truncation $\sum_{0\leq k\leq K} f_k z^k$ at a not too large order K. We then reduce to the classical problem of computing zeros of polynomials. A survey of this problem is given in [Cou1, §5.3].

The second problem (finding the subspace of functions vanishing at given points) boils down to finding the kernel of the matrix having entries the values of the functions in the chosen basis of $\mathcal{L}$ at the given points.

The only difficulty then is to control the conditioning of these two problems. This is done in two steps. We first prove [Cou1, §5.4] that the zeros of a holomorphic function on a closed disk are well conditioned unless this function is small everywhere on this disk. We then prove [Cou2, §12.7] that the form we consider cannot be small everywhere on any of the charts we consider, unless it has very small coordinates in the chosen basis of $\mathcal{L}$.

The resulting algorithm for computing in the group of divisor classes of modular curves is polynomial time in the genus and the required direct accuracy [Cou2, Theorem 12.9.1]. By *direct accuracy* we mean that the error is measured in the target space of the integration map, namely the torus $\mathbb{C}^g/\Lambda$. Saying that the direct accuracy is bounded from above by ϵ means that the returned divisor $R' = R'_1 + \cdots + R'_g$ is such that

$$\phi_b(R' - R) = \phi_b^{(g)}(R') - \phi_b^{(g)}(R)$$

is bounded from above by ϵ for the maxnorm in $\mathbb{C}^g$. This does not necessarily imply that the R_j are close to the R'_j. Indeed, in case $R = R_1 + \cdots + R_g$ is special, there exists a non-trivial linear pencil of divisors R' such that $\phi_b^{(g)}(R') = \phi_b^{(g)}(R)$. Controlling the distance between R and R' will only be possible in some cases.

In the special case when X is the curve given by equation (2.0.1) the map $\phi_b^{(1)} : X \to J$ is an isomorphism because the genus is 1. Computing with divisor classes is then very simple and the Brill-Noether algorithm takes a simple form. The space $\mathcal{L}$ consists of all degree 1 homogeneous forms, and a basis for it is made of the three projective coordinates X, Y and Z. Given P and Q, one considers the unique projective line Δ_1 through P and Q. In case $P = Q$ we take Δ_1 to be the tangent to X at P. The line Δ_1 meets X at three points: P, Q and a third point that we call S. We consider the unique projective line Δ_2 through S and the origin b. The line Δ_2 meets X at three points: b, S and a third point that we call R. On can easily check that $P + Q$ is linearly equivalent to $b + R$ or equivalently $P - b + Q - b$ is equivalent to $R - b$. The coordinates of R can be computed using very simple formulae [Sil, Chapter III]. We illustrate this using the [PARI] system. We call P the point $[0 : 0 : 1]$. We first compute Q such that $Q - b$ is linearly equivalent to $2(P - b)$. We write $Q - b \equiv 2(P - b)$ using the $\equiv$ symbol for linear equivalence. We then compute R such that $R - b \equiv P - b + Q - b \equiv 3(P - b)$. We then compute S such that $S - b \equiv Q - b + R - b \equiv 5(P - b)$.

```
>P=[0,0];
>Q=elladd(X,P,P)
[1, 1]
>R=elladd(X,P,Q)
[1, 0]
>S=elladd(X,Q,R)
[0]
```

The answer for S means that S is just the origin $b = [0 : 1 : 0]$. So the divisor $P - b$ has order 5 in the Picard group $\mathrm{Pic}(X)$, the group of divisors modulo linear equivalence.

2.3. The direct Jacobi problem. Given a divisor on X we want to compute its image by ϕ_b in the complex torus $J(\mathbb{C}) = \mathbb{C}^g/\Lambda$. It suffices to explain what to do when the divisor consists of a single point P. For every $1 \leq k \leq g$ we then have to compute $\int_b^P \omega_k$. So we integrate ω_k along any path from b to P. We split the chosen path in several pieces according to the various charts in our atlas for the Riemann surface $X(\mathbb{C})$. On every chart, the differentials ω_k can be expressed in terms of the local coordinate. We then reduce to computing integrals of the form $\int_0^{\frac{1}{2}} f(z)dz$ where $f(z)$ is holomorphic on the unit disk. Such an integral can be computed term by term. When X is a modular curve, we have a convenient system of charts and a basis for $\mathcal{L}$ consisting of forms having small coefficients in their expansions at every chart. There is long standing tradition with stating and proving bounds for these coefficients. It culminates with the so-called Ramanujan conjecture. This conjecture was proved by Deligne as a consequence of [Del1] and his proof of the

analog of the Riemann hypothesis in the Weil conjectures in [Del2]. In case X is the elliptic curve given by equation (2.0.1) we take for P the point $[0:0:1]$ and find that

$$\phi_b^{(1)}(P) = \int_b^P \omega = 2.538418608559106843377589234 \bmod \Lambda.$$

This integral is computed using the [PARI] system.

```
> ellpointtoz(X,[0,0])
2.538418608559106843377589234
```

We notice that

$$\phi_b^{(1)}(P) = \frac{2\Omega_1}{5} \bmod \Lambda.$$

So $5(P-b)$ is a principal divisor as already observed at the end of section 2.3.

2.4. The inverse Jacobi problem. At this point we have two different ways of representing a degree zero class of equivalence of divisors. We can be given a divisor in this class like the one in equation (2.2.1). Such a divisor will be called a *reduced divisor*. We can also be given a vector in $\mathbb{C}^g$ modulo the lattice of periods Λ. It is of course very easy to compute with such vectors. We also have seen in section 2.2 how to compute with reduced divisors. So both representations are convenient for computational purposes. We also have seen in section 2.3 how to pass from a reduced divisor to the corresponding point in the torus $\mathbb{C}^g/\Lambda$ applying the Abel-Jacobi map. We now consider the inverse problem: given a point $\alpha \bmod \Lambda$ in the torus $\mathbb{C}^g/\Lambda$, find some $P = P_1 + \cdots + P_g$ such that the reduced divisor $P - gb$ is mapped onto $\alpha \bmod \Lambda$ by ϕ_b.

Using an iterative method We can try an iterative method like the secant's method. We illustrate the secant's method in case X is the curve given by equation (2.0.1) and

(2.4.1) $$\alpha = (\Omega_1 + \Omega_2)/11 = 0.2884566600635348685656 + 0.1326196924489541117573i.$$

Starting from $P_0 = (50-50i, -223.147+547.739i)$ and $P_1 = (20-20i, -54.587+137.965i)$ we obtain an approximation up to 10^{-26} after eighteen iterations. We use the [PARI] system and declare a function for the secant method.

```
>secant(alpha,P0,P1,K)=
{
local(f0,f1,x0,x1,x2,P2,P3);
for(k=1,K,
f0=ellpointtoz(X,P0)-alpha;f1=ellpointtoz(X,P1)-alpha;
x0=P0[1];x1=P1[1];
x2=x1-f1*(x1-x0)/(f1-f0);
P2=[x2,ellordinate(X,x2)[1]];P3=[x2,ellordinate(X,x2)[2]];
if(abs(P2[2]-P0[2])> abs(P3[2]-P0[2]) ,P2=P3,);
P0=P1;P1=P2;
);
return(P2);
}
```

The four parameters of this function are the target point in $\mathbb{C}/\Lambda$, the two initial approximate values of P, and the number of iterations. We then type

```
>alpha=(omega1+omega2)/11;
x0=50-50*I;x1=20-20*I;
P0=[x0,ellordinate(X,x0)[2]];P1=[x1,ellordinate(X,x1)[2]];
secant(alpha,P0,P1,18)
```

Below are the results of iterations 14 to 18. We only give the values taken by the x-coordinate.

```
6.796891402429021881380876803 - 7.525836023544396684018482041i
6.796539495414535904114103146 - 7.525907619429540863361002543i
6.796539142100022043003057330 - 7.525908029913269174706910680i
6.796539142094915910541452272 - 7.525908029899464322147329306i
6.796539142094915911068237206 - 7.525908029899464321854796862i
```

The continuation method Iterative methods only work if the starting approximation is close enough to the actual solution. Such an initial approximation can be provided by the solution of a different though close inverse problem. Coming back to our example, we will start from any point on X. Say $P_0 = (0, 0)$. We compute the image $\alpha_0 \bmod \Lambda$ of P_0 by the integration map. We then choose any P_{-1} that is close enough to P_0.

```
>P0=[0,0];
alpha0=ellpointtoz(X,P0);
Pm1=[0.1,ellordinate(X,0.1)[2]];
```

We now move slowly from α_0 to α. We set $\alpha_1 = \alpha_0 + 0.1(\alpha - \alpha_0)$ and we solve the inverse problem for α_1 using the secant's method with initial values P_{-1} and P_0.

```
>P1=secant(alpha0+0.1*(alpha-alpha0),Pm1, P0,5)
[0.218773824415936734050679268 - 0.0122309960881052801981765895*I,
0.0388323642082357612959944279 - 0.00390018046133107189481433241*I]
```

We now set $\alpha_2 = \alpha_0 + 0.2(\alpha - \alpha_0)$ and we solve the inverse problem for α_2 using the secant's method with initial values P_0 and P_1.

```
>P2=secant(alpha0+0.2*(alpha-alpha0),P0, P1,5)
[0.410237833586311839505201998 - 0.0205989424813431290064696558*I,
0.111775424533436210193603161 - 0.00838376796781394064004855129*I]
```

We continue until we reach α

```
>P3=secant(alpha0+0.3*(alpha-alpha0),P1, P2,5);
                     ...
P9=secant(alpha0+0.9*(alpha-alpha0),P7, P8,5);
P10=secant(alpha,P8, P9,10)
[6.796539142094915911068237205 - 7.525908029899464321854796861*I,
-8.05657776742775028742861296 + 30.05694612451787404370259256*I]
```

This continuation method is very likely to succeed provided the integration map has a nice local behaviour all along the path from α_0 to α. This is how practical computations have been realised in [Bos1] for modular curves. It is however difficult to prove that this method works because the integration map $\phi_b^{(g)}$ has a singular locus as soon as $g > 1$, and we do not know how to provably and efficiently find a path from $\alpha_0 \bmod \Lambda$ to $\alpha \bmod \Lambda$ that keeps away from the singular locus.

3. Provably solving the inverse Jacobi problem

We have presented in section 2.4 a heuristic algorithm for the inverse Jacobi problem. This algorithm is based on continuation. It seems difficult to prove it however because that would require a good control on the singular locus of the Jacobi map. In this section we present the algorithm introduced in [Cou2]. This algorithm only requires a good control of the Jacobi map locally at a chosen divisor in $X^{(g)}$. This is a much weaker condition and it is satisfied for modular curves. An important feature of this algorithm is the use of fast exponentiation rather than continuation. The principle of fast exponentiation is recalled in section 3.1. The algorithm for

the inverse Jacobi problem itself is given in section 3.2. Section 3.3 sketches the proof of this algorithm. Proving in this context means proving the existence of a Turing machine that returns a correct answer in a given time. One has to prove both the correctness of the result and a bound for the running time. This bound here will be polynomial in the genus of the curve and the required accuracy of the result.

3.1. Fast exponentiation in groups. Assume we are given a group G. The group law in G will be denoted multiplicatively. We assume that G is *computational*. This means that we know how to represent elements in G, how to compare two given elements, how to invert a given element, and how to multiply two given elements.

The exponentiation problem in G is the following: we are given an element g in G and an integer $e \geq 2$, and we want to compute g^e as an element in G. A first possibility would be to set $a_1 = g$ and to compute $a_k = a_{k-1} \times g$ for $2 \leq k \leq e$. This requires $e - 1$ multiplications in G. It is well known, however, that we can do much better. We write the expansion of e in base 2,

$$e = \sum_{0 \leq k \leq K} \epsilon_k 2^k,$$

and we set $b_0 = g$ and $b_k = b_{k-1}^2$ for $1 \leq k \leq K$. We then notice that

$$g^e = \prod_{0 \leq k \leq K} b_k^{\epsilon_k}.$$

So we can compute g^e at the expense of a constant times $\log e$ operations in G. The algorithm above is called *fast exponentiation* and it admits many variants and improvements [Gor]. Its first known occurrence dates back to Piṅgala's Chandah-sûtra (before -200). See [DatSin, I,13].

3.2. Solving the Jacobi inverse problem by linear algebra. Recall that we have two different ways of representing an equivalence class of divisors of degree zero: reduced divisors or classes in the torus $\mathbb{C}^g/\Lambda$. We have seen that both models are computational. The Abel-Jacobi map $\phi_b^{(g)} : X^{(g)} \to \mathbb{C}^g/\Lambda$ is computational also. We want to invert it (although we know it is not quite injective). More precisely we assume we are given some α in $\mathbb{C}^g$ and we look for a degree g effective divisor on X such that $\phi_b^{(g)}(P) = \phi_b(P - gb) = \alpha \bmod \Lambda$. It seems difficult to prove the heuristic methods given in section 2.4 for this purpose. So we present here a variant for which we can give a proof, at least when X is a modular curve X_ℓ. We illustrate this method in the case where X is the curve given in equation (2.0.1). We still aim at the α given in equation (2.4.1).

We need a non-special effective divisor P_0 of degree g. Since $g = 1$ we can take any point on X. For example $P_0 = (0, 0)$. We note that the affine coordinate x is a local parameter at P_0. We choose a small real number ϵ. The smaller ϵ the better the precision of the final result. Here we choose $\epsilon = 0.0001$. We consider two points P_1 and P_2 that are very close to P_0. The first point P_1 is obtained by adding ϵ to the x-coordinate of P_0. The second point P_2 is obtained by adding ϵi to the x-coordinate of P_0.

```
P0=[0,0];
P1=[0.0001,ellordinate(X,0.0001)[2]];
P2=[0.0001*I,ellordinate(X,0.0001*I)[2]];
```

We now compute the image $\alpha_1 \bmod \Lambda$ of $P_1 - P_0$ by the Abel-Jacobi map. We also compute the image $\alpha_2 \bmod \Lambda$ of $P_2 - P_0$. We note that $\alpha_1 \bmod \Lambda$ is very close to $0 \in \mathbb{C}/\Lambda$. This is because P_0 and P_1 are close. We assume that α_1 is the smallest complex number in its class modulo Λ. We make the same assumption for α_2. Then α_1 and α_2 are two small complex numbers, and they form an $\mathbb{R}$-basis of $\mathbb{C}$. This is because the integration map $\phi_b^{(g)}$ is a local diffeomorphism at P_0 (or equivalently P_0 is a non-special divisor) and ϵ has been chosen small enough.

```
alpha1=ellpointtoz(X,P1)-ellpointtoz(X,P0);
alpha2=ellpointtoz(X,P2)-ellpointtoz(X,P0)-omega1-omega2;
```

Recall that our target in the torus $\mathbb{C}/\Lambda$ is $\alpha \bmod \Lambda$ where α is the complex number given in equation (2.4.1). So we compute the two real coordinates of α in the basis (α_1, α_2).

```
>M=[real(alpha1), real(alpha2); imag(alpha1), imag(alpha2)];
coord=M^(-1)*[real(alpha),imag(alpha)]~
[-2884.566581407009845250155464, -1326.196933330853847302268151]~
```

We deduce that α is very close to $\alpha' = -2884\alpha_1 - 1326\alpha_2$. And the class $\alpha' \bmod \Lambda$ is the image by ϕ_b of $-2884(P_1 - P_0) - 1326(P_2 - P_0)$. The linear equivalence class of the latter divisor is therefore a good approximation for our problem. There remains to compute a *reduced divisor* $P - gb$ in this class using the methods presented in section 2.2. Since the integers 2884 and 1326 are rather big, we use the fast exponentiation algorithm presented in section 3.1.

```
>coord=truncate(coord)
[-2884, -1326]~
>D1=ellsub(X,P1,P0);D2=ellsub(X,P2,P0);
P=elladd(X,ellpow(X,D1,coord[1]),ellpow(X,D2,coord[2]))
[6.798693122986621316758396123 - 7.528977879167267357619566769*I,
-8.059779911380488392224788509 + 30.07437308400090422713306570*I]
```

We now check that the image of $P - P_0$ by ϕ_b is close to α

```
>ellpointtoz(P)
0.2884000018811813146007079855 + 0.1325999988977252987328424662*I
>alpha
0.2884566600635348685656351402 + 0.1326196924489541117573536012*I
```

For a better approximation we should start with a smaller ϵ.

3.3. Matter of proof. The main concern when proving the algorithm in section 3.2 is to prove that we can find an initial divisor P_0 that is non-special. In fact we must guarantee a quantified version of this non-speciality condition. The differential of $\phi_b^{(g)}$ at P_0 should be non singular and its norm should not be too small. We can prove that such a condition holds true for modular curves [Cou2, §12.6.7] because we have a very sharp description of these curves in the neighbourhood of the points called *cusps*. As a consequence we prove [Cou2, Theorem 12.10.5] that the inverse Jacobi problem for modular curves can be solved in deterministic polynomial time in the genus and the required *direct accuracy*. Recall that *direct accuracy* means that the error is measured in the target space $\mathbb{C}^g/\Lambda$. The main difference between the algorithm in this section and the one in section 2.4 is that we only need here to control the local behaviour of $\phi_b^{(g)}$ at P_0 while the algorithm in section 2.4 requires that the map $\phi_b^{(g)}$ be non-singular above the whole path from α_0 to α.

In some cases it will be desirable to control the *inverse error* that is the error on the output divisor P in $X^{(g)}$. This will be possible when we can prove that $\phi^{(g)} : X^{(g)} \to J$ is a local diffeomorphism at P (that is P is non-special). We will also need a lower bound for the norm of the differential of $\phi^{(g)}$ at P. Such a lower bound can be provided by arithmetic.

4. Computing torsion points I

In this and the next section we will assume that X is a modular curve and ℓ a prime number. We will be interested in ℓ-torsion points in the torus $J(\mathbb{C}) = \mathbb{C}^g/\Lambda$. A point

$$a = \alpha \bmod \Lambda$$

is an ℓ-torsion point if and only if α lies in $\frac{1}{\ell}\Lambda$. So the ℓ-torsion subgroup of $J(\mathbb{C})$ is $\frac{1}{\ell}\Lambda/\Lambda$ and it has cardinality ℓ^{2g}. This group is also denoted $J[\ell]$.

Some of these torsion points carry important arithmetic information. The values taken by algebraic functions at these points generate interesting number fields. We want to compute these fields. In this section we will focus on a special case. We will assume that X is the genus 1 curve given in equation (2.0.1) and $\ell = 11$. A more general situation will be studied in the next section 5. We notice that the curve in equation (2.0.1) is indeed the modular curve known as X_{11}. Since X has genus 1, the map $\phi_b : X \to J$ is an isomorphism mapping $b = [0:1:0]$ onto the origin. So the affine coordinate x and y induce algebraic functions $x \circ \phi_b^{-1}$ and $y \circ \phi_b^{-1}$ on J. There are $11^2 = 121$ points of 11 torsion in J and 0 is one of them. We will be interested in the values taken by $x \circ \phi_b^{-1}$ at the remaining 120 points of 11-torsion. On can check that $x \circ \phi_b^{-1}$ takes the same value at two opposite points. So there only remain 60 values of interest. These are algebraic numbers and they form a single orbit under the action of the Galois group $\mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$. So it is natural to consider their annihilating polynomial

$$H(T) = \prod_{0 \neq a \in J[11]/\pm 1} \left(T - x(\phi^{-1}(a))\right). \tag{4.0.1}$$

This is an irreducible polynomial in $\mathbb{Q}[T]$. Computing such polynomials is a cornerstone in the algorithmic of modular forms and Galois representations.

4.1. An algebraic approach. The polynomial in equation (4.0.1) is known as the 11-th division polynomial ψ_{11} of the genus one curve X. For every $k \geq 1$ one can define the k-th division polynomial $\psi_k(T)$ to be the annihilating polynomial of the x-coordinates of all non-zero k-torsion points on X. These polynomials can be computed using recursion formulae [Eng, Section 3.6] [Sil, Exercise 3.7] that follow from the simple algebraic form of the addition law on X. Using these recursion formulae we find

$$H(T) = T^{60} - 20T^{59} + 112T^{58} + 1855T^{57} + \cdots + 1321T^4 - 181T^3 + 22T^2 - 2T + 1/11.$$

So we have an efficient algebraic method to compute $H(T)$. We will explain in section 5 why it seems difficult to us to generalize this algebraic method to curves of higher genus.

4.2. Using complex approximations. In this section we compute complex approximations of the coefficients of $H(T)$. We also explain how one can deduce the exact value of these coefficients from a sharp enough complex approximation. We have seen in sections 2.4 and 3 how to invert the map ϕ_b. Given a point a in the torus $\mathbb{C}^g/\Lambda$ we can compute a complex approximation of some reduced divisor $P_a - gb$ such that $\phi_b(P_a - bg) = a$. Since here the genus is one, P_a consists of a single point on X, and it is uniquely defined. In case $a = (\Omega_1 + \Omega_2)/11$ we already found that the x-coordinate $x(P_a)$ of P_a is

$$6.7965391420949159110682372 06 - 7.5259080298994643218547968 62i$$

up to an error of 10^{-27}. We let a run over the 60 elements in $(J[11] - \{0\})/\pm 1$ and compute the 60 corresponding values of $x(P_a)$ with the same accuracy. We then compute their sum and find it is equal to 20 up to an error of 10^{-25}. This suggests that the coefficient of T^{59} in $H(T)$ is -20. In order to turn this heuristic into a proof, we need some information about the coefficients of $H(T)$. We know that these coefficients are rational numbers. We need an upper bound on their *height*. The *height* of a rational number is the maximum of the absolute values of its numerator and denominator. We explain in the next section 4.3 how a good approximation and a good bound on the height suffice to characterise and compute a rational number. In case X is the curve given in equation (2.0.1) an upper bound on the height of the coefficients of $H(T)$ can be proved by elementary means. For example we know that the denominator of these coefficients is either 1 or 11. In case X is a modular curve, similar bounds will be necessary. These bounds have been proved by the second author in collaboration with de Jong in [Ed-Jo1] and [Ed-Jo2], using Arakelov theory and arithmetic geometry together with a result of Merkl in [Merkl] on upper bounds for Green functions.

All the coefficients of $H(T)$ are computed in the same way. They are symmetric functions of the $x(P_a)$, so we can compute sharp approximations for them. We deduce their exact values using an a priori bound on their height.

4.3. Recovering a rational number from a good approximation. In the previous section 4.2 we claimed that a rational number $x = a/b$ can be recovered from a sharp enough complex approximation, provided we have an a priori bound on the height of x. We recall that the height of a rational number a/b, with a and b integers that are relatively prime, is $\max\{|a|, |b|\}$. The rational number $x = a/b$ is known if we know an upper bound h for its height and an approximation y of it (in $\mathbb{R}$, say), with $|x-y| < 1/(2h^2)$. Indeed, if $x' = a'/b'$ also has height at most h, and $x' \neq x$, then

$$|x - x'| = \left|\frac{a}{b} - \frac{a'}{b'}\right| = \left|\frac{ab' - ba'}{bb'}\right| \geq \frac{1}{|bb'|} \geq 1/h^2.$$

We also note that there are good algorithms to deduce x from such a pair of an approximation y and a bound h, for example, by using continued fractions, as we will now explain.

In practice we will use rational approximations y of x. Every rational number y can be written uniquely as

$$[a_0, a_1, \ldots, a_n] = a_0 + \cfrac{1}{a_1 + \cfrac{1}{\ddots \; a_{n-1} + \cfrac{1}{a_n}}},$$

where $n \in \mathbb{Z}_{\geq 0}$, $a_0 \in \mathbb{Z}$, $a_i \in \mathbb{Z}_{>0}$ for all $i > 0$, and $a_n > 1$ if $n > 0$. To find these a_i, one defines $a_0 := \lfloor y \rfloor$ and puts $n = 0$ if $y = a_0$; otherwise, one puts $y_1 := 1/(y - a_0)$ and $a_1 = \lfloor y_1 \rfloor$ and $n = 1$ if $y_1 = a_1$, and so on. The rational numbers $[a_0, a_1, \ldots, a_i]$ with $0 \leq i \leq n$ are called the *convergents* of the continued fraction of y. Then one has the following well-known result (see Theorem 184 from [Ha-Wr]).

Proposition 4.3.1. *Let y be in $\mathbb{Q}$, a and b in $\mathbb{Z}$ with $b \neq 0$, and*

$$\left|\frac{a}{b} - y\right| < \frac{1}{2b^2}.$$

Then a/b is a convergent of the continued fraction of y.

5. Computing torsion points II

In this section we describe how we compute the fields of definition of certain torsion points in Jacobians of modular curves. We recommend [Di-Sh] to those who are interested in an introduction to the theory of modular forms.

Let $\mathrm{SL}_2(\mathbb{Z})$ denote the group of 2 by 2 matrices with coefficients in $\mathbb{Z}$ and with determinant one. It acts on the complex upper half plane $\mathbb{H}$ via fractional linear transformations

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot z = \frac{az + b}{cz + d}. \tag{5.1}$$

The standard fundamental domain F for $\mathrm{SL}_2(\mathbb{Z})$ acting on $\mathbb{H}$ (see Figure 1) consists of the z with $|z| \geq 1$ and $|\Re(z)| \leq 1/2$. It is not bounded, hence not compact. Viewing $\mathbb{H}$ as the open northern hemisphere in $\mathbb{P}^1(\mathbb{C}) = \mathbb{C} \cup \{\infty\}$, with boundary the equator $\mathbb{P}^1(\mathbb{R})$, we see that the closure $\overline{F}$ of F in $\mathbb{P}^1(\mathbb{C})$ is the union of F and the point ∞.

For every prime number ℓ we let Γ_ℓ denote the subset of $\mathrm{SL}_2(\mathbb{Z})$ consisting of the $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ with c, $a - 1$ and $d - 1$ divisible by ℓ. Then Γ_ℓ is a subgroup of $\mathrm{SL}_2(\mathbb{Z})$, of index $\ell^2 - 1$. We assume that $\ell \geq 5$ from now on. Then the action of Γ_ℓ on $\mathbb{H}$ is free. Each z in $\mathbb{H}$ has a neighbourhood U

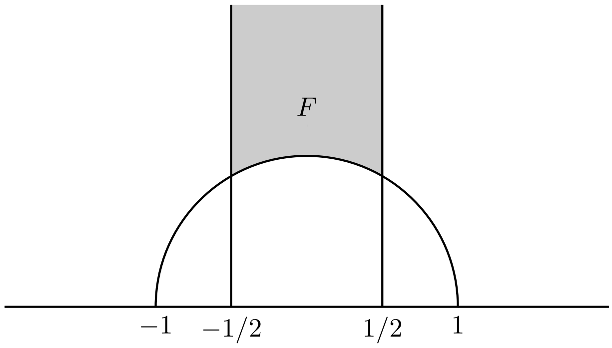

Figure 1. Standard fundamental domain F for $\mathrm{SL}_2(\mathbb{Z})$ acting on $\mathbb{H}$

such that all γU for γ in Γ_ℓ are disjoint. The quotient $\Gamma_\ell\backslash\mathbb{H}$ is therefore a Riemann surface that we denote by Y_ℓ, and the quotient map $\mathbb{H} \to Y_\ell$ is a covering map, that is, each point y in Y_ℓ has an open neighbourhood U such that the inverse image of U in $\mathbb{H}$ is the disjoint union of copies of U, indexed by the inverse image of y.

The Riemann surface Y_ℓ is not compact. A fundamental domain F_ℓ in $\mathbb{H}$ for Γ_ℓ can be gotten as the union of the γF, where γ ranges over a set of representatives of $\Gamma_\ell\backslash\,\mathrm{SL}_2(\mathbb{Z})/\{1,-1\}$. Such a set consists of $(\ell^2-1)/2$ elements and it can easily be found. We can compactify Y_ℓ to a compact Riemann surface X_ℓ by adding $\ell-1$ points, called *cusps*, the points of $\mathbb{P}^1(\mathbb{R})$ that lie in the closure of F_ℓ in $\mathbb{P}^1(\mathbb{C})$. These points lie in fact in $\mathbb{P}^1(\mathbb{Q})$ and can easily be written down. All this leads to an explicit topological and analytic description of X_ℓ. It is covered by coordinate disks around the cusps. For example, the function

$$q\colon \mathbb{H} \to \mathbb{C}, \quad z \mapsto e^{2\pi i z}, \tag{5.2}$$

restricted to the set of z with $\Im(z) > 1/\ell$, induces a coordinate on a disk in X_ℓ around the cusp ∞. Indeed, the image under q of this region is the punctured disk of radius $e^{-2\pi/\ell}$ around 0, and the cusp ∞ fills the puncture. The genus g_ℓ of X_ℓ is equal to $(\ell-5)(\ell-7)/24$. For $\ell = 11$ the genus is 1, and indeed, X_{11} is the elliptic curve X_{11} given by equation (2.0.1).

It is of course a miracle that such an analytically defined Riemann surface as X_{11} is defined over $\mathbb{Q}$, that is, can be described as a curve in a projective space given by a equations with coefficients in $\mathbb{Q}$. But this is true for all ℓ, and it is explained as follows, for $\ell > 13$. The theory of modular forms gives that the $\mathbb{C}$-vector spaces $\Omega^1(X_\ell)$ of holomorphic differentials on X_ℓ have bases consisting of 1-forms ω whose pullback to $\mathbb{H}$ is of the form $(\sum_{n\geq 1} a_n q^n)\cdot(dq)/q$ with all a_n in $\mathbb{Z}$. Quotients of such ω and ω' in $\Omega^1(X_\ell)$ then provide sufficiently many rational functions on X_ℓ to embed it into a projective space, such that the image is given by homogeneous polynomial equations with coefficients in $\mathbb{Q}$.

We let J_ℓ denote the Jacobian variety of X_ℓ. It is also defined over $\mathbb{Q}$, as well as its group law. This means that the group law is described by quotients of polynomials with coefficients in $\mathbb{Q}$. Therefore, for all P and Q in J_ℓ and for each σ in $\mathrm{Aut}(\mathbb{C})$, the automorphism group of the field $\mathbb{C}$, we have $\sigma(P+Q) = \sigma(P) + \sigma(Q)$. For each integer $m \geq 1$ the kernel $J_\ell[m]$ of the multiplication by m map is finite (it consists of m^{2g_ℓ} elements) and preserved by the action of $\mathrm{Aut}(\mathbb{C})$. This implies that all P in $J_\ell[m]$ have coordinates in the algebraic closure $\overline{\mathbb{Q}}$ of $\mathbb{Q}$ in $\mathbb{C}$, that is, for each rational function f on J_ℓ that is defined over $\mathbb{Q}$ and has no pole at P, the value $f(P)$ of f at P is in $\overline{\mathbb{Q}}$. The analytic description above of X_ℓ gives us an analytic description of J_ℓ.

We are interested in certain subgroups V_ℓ of the ℓ-torsion subgroup $J_\ell[\ell]$ of J_ℓ that are invariant under the Galois group $\mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$ and consist of ℓ^2 elements. These V_ℓ can be described explicitly and efficiently in terms of certain operators called Hecke operators on the first homology group of X_ℓ. The whole point is to understand them algebraically, with their $\mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$-action.

The subgroup V_ℓ defines a commutative $\mathbb{Q}$-algebra A_ℓ of dimension ℓ^2 as $\mathbb{Q}$-vector space, the

coordinate ring of V_ℓ over $\mathbb{Q}$. This algebra A_ℓ consists of the functions $f\colon V_\ell \to \overline{\mathbb{Q}}$ with the property that for all σ in $\mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$ and all P in V_ℓ we have $f(\sigma(P)) = \sigma(f(P))$. Addition and multiplication are pointwise. Each f_ℓ in A_ℓ with the property that the $f_\ell(P)$ are all distinct is a generator, and A_ℓ is then given as $\mathbb{Q}[T]/(H_{f_\ell})$, with

$$H_{f_\ell} = \prod_{P \in V_\ell} (T - f_\ell(P)) \quad \text{in } \mathbb{Q}[T].$$

A direct approach for computing A_ℓ or H_{f_ℓ} algebraically, as in Section 4.1 in the case of the division polynomial ψ_{11}, is very unlikely to succeed in time polynomial in ℓ, because in the case of V_ℓ one has to work with the algebraic variety J_ℓ, whose dimension grows quadratically with ℓ. Writing down polynomial equations with coefficients in $\mathbb{Q}$ for J_ℓ and V_ℓ is probably still possible, in time polynomial in ℓ. But computing a $\mathbb{Q}$-basis of A_ℓ from the equations in a standard way uses Groebner basis methods, which, as far as we know, take time exponential or even worse in the number of variables, that is, exponential or worse in ℓ.

For this reason we replace, in [BCEJM], exact computations by approximations. There are then two problems to be dealt with. The first is to show that f_ℓ can be chosen so that the logarithm of the height of the coefficients of H_{f_ℓ}, that is, the number of digits of their numerator and denominator, does not grow faster than a power of ℓ. This problem is solved in [Ed-Jo1], [Merkl] and [Ed-Jo2], using arithmetic algebraic geometry and analysis on Riemann surfaces. The second problem is to show that for the same choice of f_ℓ, the values $f_\ell(P)$ at all P in V_ℓ can be approximated in $\mathbb{C}$ with a precision of n digits in time polynomial in $n+\ell$. This is done in [Cou2]. The chapters [Bos1] and [Bos2] contain real computations using the method of Section 2.4, for prime numbers $\ell \leq 23$.

Let us now explain how we choose f_ℓ (up to some technicalities; the precise setup is given in [Ed3, §8.2]) and say some words about the approximation of the $f_\ell(P)$. Standard functions on Jacobian varieties such as J_ℓ are theta functions. But a problem is that these are usually given as power series in g_ℓ variables, and as g_ℓ grows this can make the number of terms that must be evaluated for a sufficiently good approximation grow exponentially in ℓ. In other words, we know no method to approximate their values fast enough (of course, it is not excluded that such methods do exist). Our solution is to transfer the problem from J_ℓ to $X_\ell^{g_\ell}$, via the Abel-Jacobi map. We choose h_ℓ a suitable non-constant rational function on X_ℓ, defined over $\mathbb{Q}$, of small degree and with small coefficients. Then we take as origin a suitable divisor of degree g_ℓ on X_ℓ, defined over $\mathbb{Q}$. This divisor is carefully chosen in [Ed3] to have the following property: for each P in V_ℓ there is a *unique* effective divisor $Q_P = Q_{P,1} + \cdots + Q_{P,g_\ell}$ on X_ℓ, such that its image under the Abel-Jacobi map is P. Then we define $f_\ell(P) = h_\ell(Q_{P,1}) + \cdots + h_\ell(Q_{P,g_\ell})$. Rather magically, the problem of power series in many variables has disappeared. The function h_ℓ is locally given by a power series in *one* variable. We evaluate it at each $Q_{P,i}$ separately. The Abel-Jacobi map (see Section 2) is given by a sum of g_ℓ integrals of g_ℓ-tuples of holomorphic 1-forms in one variable. The analytic description above of X_ℓ and J_ℓ should make it clear that the Abel-Jacobi map and the function h_ℓ can be well approximated with standard tools. That means that the only remaining problem is the inversion of the Abel-Jacobi map, that is, the approximation of the divisors Q_P, but that was discussed and solved in Sections 2 and 3. The main result obtained in [BCEJM] is the following theorem.

Theorem 5.3. *There is a deterministic algorithm that on input a prime number $\ell \geq 11$ computes the $\mathbb{Q}$-algebra A_ℓ in time polynomial in ℓ.*

A *probabilistic* algorithm for computing A_ℓ is also given in [BCEJM]. It relies on p-adic approximations rather than complex approximations. In [Cou3] it is explained how such p-adic approximations can be computed efficiently. From a theoretical point of view, a probabilistic algorithm is not quite as satisfactory as a deterministic one. From a practical point of view, it is just as good. In our case the probabilistic algorithm has a simpler proof than the deterministic one. And Peter Bruin [Bru1, Bru2] has been able to generalize it to a much wider class of V_ℓ-like modular spaces. Finding a similar generalization for the deterministic algorithm is an open problem at this time.

6. Applications and open questions

The main motivation for all the work done in [BCEJM] is the application in number theory to the fast computation of coefficients of modular forms. Instead of attempting to present this in the most general case we give two examples: Ramanujan's τ-function, and powers of Jacobi's θ-function.

Recall that $q\colon \mathbb{H} \to \mathbb{C}$ is the function $z \mapsto e^{2\pi i z}$. The *discriminant modular form* Δ is the holomorphic function on $\mathbb{H}$ given by the converging infinite product

$$\Delta = q \prod_{n\geq 1}(1-q^n)^{24}. \tag{6.1}$$

The holomorphic function Δ has a power series expansion in q,

$$\Delta = \sum_{n\geq 1} \tau(n) q^n, \tag{6.2}$$

whose coefficients, which are integers, define *Ramanujan's τ-function.* It can be shown that for all $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ in $\mathrm{SL}_2(\mathbb{Z})$, and for all z in $\mathbb{H}$, we have

$$\Delta\left(\frac{az+b}{cz+d}\right) = (cz+d)^{12}\Delta(z). \tag{6.3}$$

Functions $f\colon \mathbb{H} \to \mathbb{C}$ that are given by a power series $\sum_{n\geq 1} a_n(f)q^n$ with this symmetry under the action of $\mathrm{SL}_2(\mathbb{Z})$ on $\mathbb{H}$ with the exponent 12 replaced by an integer k are called cuspidal modular forms of weight k on $\mathrm{SL}_2(\mathbb{Z})$. The complex vector spaces $S(\mathrm{SL}_2(\mathbb{Z}), k)$ of cuspidal modular forms of weight k are finite dimensional. The dimension grows roughly as $k/12$. More precisely, for $k < 12$ the space $S(\mathrm{SL}_2(\mathbb{Z}), k)$ is zero, and $S(\mathrm{SL}_2(\mathbb{Z}), 12)$ is one-dimensional, generated by Δ. The fact that each g in $\mathrm{GL}_2(\mathbb{Q})$ with $\det(g) > 0$ acts on $\mathbb{H}$ and normalises $\mathrm{SL}_2(\mathbb{Z})$ up to finite index leads to operators $T_{k,g}$ on the $S(\mathrm{SL}_2(\mathbb{Z}), k)$. These operators are named after Hecke. For each integer $n \geq 1$ there is an operator $T_{k,n}$; for n prime it is the one induced by the matrix $\left(\begin{smallmatrix} n & 0 \\ 0 & 1 \end{smallmatrix}\right)$ and for general n it is a bit more complicated. As the space $S(\mathrm{SL}_2(\mathbb{Z}), 12)$ is one-dimensional, each $T_{12,n}$ acts on it as multiplication by a scalar. This scalar turns out to be the coefficient $\tau(n)$ of q^n in the power series of Δ. Well known relations between the Hecke operators imply relations between the $\tau(n)$ that are summarised in the identity of Dirichlet series, for s in $\mathbb{C}$ with real part $\Re(s)$ large enough:

$$\sum_{n\geq 1} \tau(n) n^{-s} = \prod_p (1 - \tau(p)p^{-s} + p^{11}p^{-2s})^{-1}. \tag{6.4}$$

Here the product is over all prime numbers, and both sides converge for $\Re(s) > 13/2$. In fact, it is a famous theorem of Deligne ([Del1] and [Del2]) that for all primes p one has

$$|\tau(p)| \leq 2p^{11/2}, \tag{6.5}$$

as conjectured by Ramanujan.

The identity of Dirichlet series shows that for n and m with greatest common divisor 1 we have $\tau(nm) = \tau(n)\tau(m)$, and that for p prime and $r \geq 2$ we have $\tau(p^r) = \tau(p)\tau(p^{r-1}) - p^{11}\tau(p^{r-2})$. Therefore, the computation of $\tau(n)$ is reduced to that of the $\tau(p)$ for p dividing n. We can now state one of the main theorems of [BCEJM].

Theorem 6.6. *There is a deterministic algorithm that on input an integer $n \geq 1$ together with its factorisation into prime factors computes $\tau(n)$ in time polynomial in $\log n$.*

Before this result, the fastest known algorithms to compute $\tau(n)$ took time exponential in $\log n$. For example, if one computes the product in (6.1) up to order n by multiplying the necessary

factors, then one spends time at least linear in n. To prove the theorem, it suffices to show that for p prime $\tau(p)$ can be computed in time polynomial in $\log p$. This will be done using Theorem 5.3.

The fact that modular forms have an enormous amount of symmetry as in (6.3) is certainly powerful, but it does not suffice at this point. What is needed is Galois symmetry, which is also behind Deligne's famous result mentioned above. A lot could be said on this, but this is not an appropriate place for that.

In a nutshell: modular forms give elements in de Rham cohomology of complex algebraic varieties defined over $\mathbb{Q}$, and the singular homology with torsion coefficients $\mathbb{Z}/\ell\mathbb{Z}$ of those complex varieties can be defined algebraically (Grothendieck, Artin, Deligne) and therefore has an action by $\mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$.

For example, Δ gives rise, for every prime integer $\ell \geq 11$, to a certain subgroup V_ℓ of the ℓ-torsion of the Jacobian J_ℓ of X_ℓ. This subgroup V_ℓ has cardinality ℓ^2. For $\ell \neq p$ the image of $\tau(p)$ in $\mathbb{Z}/\ell\mathbb{Z}$ is determined by the action of $\mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$ on this V_ℓ. The addition map $V_\ell \times V_\ell \to V_\ell$ induces a $\mathbb{Q}$-algebra morphism called co-addition from A_ℓ to $A_\ell \otimes A_\ell$, that is, from $\mathbb{Q}[T]/(H_{f_\ell})$ to $\mathbb{Q}[T_1, T_2]/(H_{f_\ell}(T_1), H_{f_\ell}(T_2))$. Computing $\tau(p)$ modulo ℓ (for $p \neq \ell$) is then done by reducing A_ℓ with its co-addition modulo p and computing on this reduction $A_{\ell,p}$ a certain relation between the co-addition and the Frobenius map that sends a in $A_{\ell,p}$ to a^p, just as in Schoof's algorithm for elliptic curves (see Section 1.2 of [Ed1, §1.2]). For more detail the interested reader is referred to Section 2.4 of [Ed2, §2.4] and the references therein. The point is that this advanced machinery can actually be used for computing $\tau(p)$ mod ℓ in time polynomial in $\log p$ and ℓ.

In order to recover the actual value of $\tau(p)$ as an integer, we compute $\tau(p)$ modulo several small primes ℓ. If the product of these small primes is bigger than $4p^{5.5}$ then we deduce $\tau(p)$ using inequality (6.5) and Chinese remainder theorem [Coh, 1.3.3].

We now come to our second example: the classical question in how many ways a positive integer n can be written as a sum of $d \geq 1$ squares of integers. Let us write $r_d(n)$ for this number, that is, $r_d(n) = \#\{x \in \mathbb{Z}^d : x_1^2 + \cdots + x_d^2 = n\}$. Then $r_d(n)$ is the coefficient of q^n in the formal power series θ_d, with

$$\theta_d = \sum_{n \geq 0} r_d(n) q^n = \sum_{x \in \mathbb{Z}^d} q^{x_1^2 + \cdots + x_d^2} = \left(\sum_{x_1 \in \mathbb{Z}} q^{x_1^2} \right) \cdots \left(\sum_{x_d \in \mathbb{Z}} q^{x_d^2} \right) = \theta_1^d \quad \text{in } \mathbb{Z}[[q]]. \tag{6.7}$$

The formal power series θ_1 defines a holomorphic complex function on the complex upper half plane $\theta \colon \mathbb{H} \to \mathbb{C}$ by viewing q as the function $q \colon z \mapsto e^{2\pi i z}$. Poisson's summation formula then shows that for all $z \in \mathbb{H}$ we have

$$\theta(-1/4z) = (-2iz)^{1/2} \theta(z), \tag{6.8}$$

where the square root is continuous and positive for z in $i {\cdot} \mathbb{R}_{>0}$. This functional equation for θ, together with the obvious one $\theta(z+1) = \theta(z)$, imply that θ is a modular form of weight $1/2$, and therefore that θ_d (interpreted as a function on $\mathbb{H}$) is a modular form of weight $d/2$.

This fact is the origin of many results concerning the numbers $r_d(n)$. The famous explicit formulas for the $r_d(n)$ for even d up to 10 due to Jacobi, Eisenstein and Liouville (see [Mil] and Chapter 20 of [Ha-Wr]) owe their existence to it. In order to state these formulas, let $\sum_{d|m}$ denote summation over the positive divisors d of m, with the convention that there are no such d if m is not an integer, and let $\chi \colon \mathbb{Z} \to \mathbb{C}$ be the map that sends n to 0 if n is even, to 1 if n is of the form

$4m+1$ and to -1 if n is of the form $4m-1$. Then we have:

$$\begin{aligned}
r_2(n) &= 4\sum_{d|n}\chi(d),\\
r_4(n) &= 8\sum_{2\nmid d|n} d + 16\sum_{2\nmid d|(n/2)} d,\\
r_6(n) &= 16\sum_{d|n}\chi\left(\frac{n}{d}\right)d^2 - 4\sum_{d|n}\chi(d)d^2,\\
r_8(n) &= 16\sum_{d|n}d^3 - 32\sum_{d|(n/2)}d^3 + 256\sum_{d|(n/4)}d^3,\\
r_{10}(n) &= \frac{4}{5}\sum_{d|n}\chi(d)d^4 + \frac{64}{5}\sum_{d|n}\chi\left(\frac{n}{d}\right)d^4 + \frac{8}{5}\sum_{d\in\mathbb{Z}[i],\,|d|^2=n} d^4.
\end{aligned}$$

In the last formula, $\mathbb{Z}[i]$ is the set of Gaussian integers $a+bi$ in $\mathbb{C}$ with a and b in $\mathbb{Z}$.

Using these formulas, the numbers $r_d(n)$ for d in $\{2,4,6,8,10\}$ can be computed in time polynomial in $\log n$, if n is given with its factorisation in prime numbers. This is not the case for the formulas that were found a bit later by Glaisher for $r_d(n)$ for some even $d \geq 12$. We give the formula that he found for $d=12$, as interpreted by Ramanujan:

$$r_{12}(n) = 8\sum_{d|n}d^5 - 512\sum_{d|(n/4)}d^5 + 16a_n, \quad \text{where} \quad \sum_{n\geq 1}a_nq^n = q\prod_{m\geq 1}(1-q^{2m})^{12}. \tag{6.9}$$

Computing a_n by multiplying out the factors $1-q^{2m}$ up to order n takes time at least linear in n, hence exponential in $\log n$. We know of no direct way to compute the a_n in time polynomial in $\log n$, even if n is given with its factorisation. However, $\sum_{n\geq 1}a_nq^n$ is a modular form, and therefore we *can* compute a_n in time polynomial in $\log n$, if n is given with its factorisation, *via* the computation of Galois representations. The same is true for the $r_d(n)$ for all even d. The explicit formulas for $d \leq 10$ correspond precisely to the cases where the Galois representations that occur are of dimension one, whereas for $d \geq 12$ genuine two-dimensional Galois representations always occur, as proved by Ila Varma in her master's thesis [Var].

We conclude that from an algorithmic perspective the classical problem of computing the $r_d(n)$ for even d and n given with its factorisation into primes is solved for *all* even d. The question as to the existence of *formulas* has a negative answer, but for *computations* this does not matter.

Open questions Finally, we should point out that the algorithms in theorems 5.3 and 6.6, despite their polynomial time complexity, are not so practical at present. However, Bosman's computation of the V_ℓ associated with Δ for ℓ in $\{13,17,19\}$ enabled him to further study Lehmer's conjecture on the values of $\tau(n)$ modulo n. See Lygeros and Rozier [Ly-Ro] for a more classical experimental approach. A challenge for the near future is to design and implement a practical variant of these algorithms.

References

[Bos1] J. Bosman, Computations with modular forms and Galois representations. In *Computational Aspects of Modular Forms and Galois Representations.* Annals of Mathematics Studies 176, Princeton University Press, Princeton, NJ, 2011, 129–157.

[Bos2] J. Bosman, Polynomials for projective representations of level one forms. In *Computational Aspects of Modular Forms and Galois Representations.* Annals of Mathematics Studies 176, Princeton University Press, Princeton, NJ, 2011, 159–172.

[BCEJM] B. Edixhoven and J.-M. Couveignes editors, with contributions by J. Bosman, J.-M. Couveignes, B. Edixhoven, R. de Jong and F. Merkl, *Computational Aspects of Modular Forms and Galois Representations.* Annals of Mathematics Studies 176, Princeton University Press, Princeton, NJ, 2011.

[Bri-Noe] A. Brill and M. Noether, Über die algebraischen Functionen und ihre Anwendung in der Geometrie. *Mathematische Annalen.* **7** (1874), 269–310.

[Bru1] P. Bruin, Modular curves, Arakelov theory, algorithmic applications. PhD-thesis, Leiden, 2010. Available on-line at: `http://hdl.handle.net/1887/15915`

[Bru2] Peter Bruin, Computing coefficients of modular forms. *Publications mathématiques de Besançon.* (2011), 19–36. Available on-line at: `http://pmb.univ-fcomte.fr/2011.html`

[Coh] H. Cohen, *A course in computational algebraic number theory.* Graduate Texts in Mathematics 138. Springer, Berlin, 1993.

[Cou1] J.-M. Couveignes, Computing complex zeros of polynomials and power series. In *Computational Aspects of Modular Forms and Galois Representations.* Annals of Mathematics Studies 176, Princeton University Press, Princeton, NJ, 2011, 95–128.

[Cou2] J.-M. Couveignes, Approximating V_f over the complex numbers. In *Computational Aspects of Modular Forms and Galois Representations.* Annals of Mathematics Studies 176, Princeton University Press, Princeton, NJ, 2011, 257–336.

[Cou3] J.-M. Couveignes, Computing V_f modulo p. In *Computational Aspects of Modular Forms and Galois Representations.* Annals of Mathematics Studies 176, Princeton University Press, Princeton, NJ, 2011, 337–370.

[Cre] J.E. Cremona, *Algorithms for modular elliptic curves.* Cambridge University Press, London, 1997.

[DatSin] B. Datta and A.N. Singh, *History of Hindu Mathematics.* Motilal Banarsi Das, Lahore, 1935.

[Del1] P. Deligne, Formes modulaires et représentations l-adiques. Séminaire Bourbaki **355** (1969).

[Del2] P. Deligne, La conjecture de Weil. I. Inst. Hautes Études Sci. Publ. Math. **43** (1974), 273–307.

[Di-Sh] F. Diamond and J. Shurman, *A first course in modular forms.* GTM 228, Springer, Berlin, 2005.

[Ed1] B. Edixhoven, Introduction, main results, context. In *Computational Aspects of Modular Forms and Galois Representations.* Annals of Mathematics Studies 176, Princeton University Press, Princeton, NJ, 2011, 1–27.

[Ed2] B. Edixhoven, Modular curves, modular forms, lattices, Galois representations. In *Computational Aspects of Modular Forms and Galois Representations.* Annals of Mathematics Studies 176, Princeton University Press, Princeton, NJ, 2011, 29–68.

[Ed3] B. Edixhoven, Description of $X_1(5l)$. In *Computational Aspects of Modular Forms and Galois Representations.* Annals of Mathematics Studies 176, Princeton University Press, Princeton, NJ, 2011, 173–185.

[Ed-Jo1] B. Edixhoven and R. de Jong, Applying Arakelov theory. In *Computational Aspects of Modular Forms and Galois Representations.* Annals of Mathematics Studies 176, Princeton University Press, Princeton, NJ, 2011, 187–201.

[Ed-Jo2] B. Edixhoven and R. de Jong, Bounds for Arakelov invariants of modular curves. In *Computational Aspects of Modular Forms and Galois Representations.* Annals of Mathematics Studies 176, Princeton University Press, Princeton, NJ, 2011, 217–256.

[Eng] A. Enge, *Elliptic curves and their applications to cryptography, an introduction.* Kluwer Academic, New York, 1999.

[Fre] G. Frey and M. Müller, Arithmetic of modular curves and applications. In *On Artin's conjecture for odd* 2*-dimensional representations.* Lecture Notes in Math. 1585, Springer, Berlin, 1994.

[Gor] D. M. Gordon, A Survey of Fast Exponentiation Methods. J. Algorithms **27**(1) (1998), 129–146.

[Ha-Wr] G.H. Hardy and E.M. Wright, *An introduction to the theory of numbers.* Fifth edition. Clarendon Press, New York, 1979.

[Ly-Ro] N. Lygeros and O. Rozier, A new solution to the equation $\tau(p) \equiv 0 \pmod p$. J. Integer Seq. **13** (2010), no. 7, Article 10.7.4, 11 pp.

[Man] Y. Manin, Parabolic points and zeta function of modular curves. Math. USSR Izvestija **6** (1972), no. 1, 19–64.

[Merel] L. Merel, Universal Fourier expansions of modular forms. In *On Artin's conjecture for odd* 2*-dimensional representations.* Lecture Notes in Math. 1585, Springer, Berlin, 1994, 59–94.

[Merkl] F. Merkl, An upper bound for Green functions on Riemann surfaces. In *Computational Aspects of Modular Forms and Galois Representations.* Annals of Mathematics Studies 176, Princeton University Press, Princeton, NJ, 2011, 203–215.

[Mil] S.C. Milne, Infinite families of exact sums of squares formulas, Jacobi elliptic functions, continued fractions, and Schur functions. Ramanujan J. **6** (2002), no. 1, 7–149.

[PARI] C. Batut, K. Belabas, D. Bernardi, H. Cohen, and M. Olivier, *User's guide to PARI/GP (version 2.3.1).* `http://pari.math.u-bordeaux.fr.`

[Sil] J. Silverman, *The arithmetic of elliptic curves.* Lecture Notes in Math. 106, Springer, Berlin, 1986.

[Ste] W.A. Stein, *Modular forms, a computational approach.* With an appendix by Paul E. Gunnells. Graduate Studies in Mathematics, 79. American Mathematical Society, Providence, RI, 2007.

[Var] I. Varma, Finding elementary formulas for theta functions associated to even sums of squares. Indag. Math. (N.S.) **22** (2011), 12-26.

[Vol] E. Volcheck, Computing in the Jacobian of a plane algebraic curve. In *Algorithmic Number Theory Conference.* Lecture Notes in Computer Sciences, volume 877, Springer 1994, 221-233,

Jean-Marc Couveignes, Univ. Bordeaux, IMB, UMR 5251, F-33400 Talence, France. CNRS, IMB, UMR 5251, F-33400 Talence, France. INRIA, F-33400 Talence, France

E-mail: Jean-Marc.Couveignes@math.u-bordeaux1.fr

Mathematisch Instituut, Universiteit Leiden, Niels Bohrweg 1, 2333 CA Leiden, Nederland

E-mail: edix@math.leidenuniv.nl